Installing build tools for GDS

Stephen Leake

23 October 2003

1 Introduction

The build tools for GDS are all Open Source tools, and are available on both Windows and Lynx.

The primary integrated development environment is Gnu Emacs. Ada Core Technologies GPS may also be used, but it is not as powerful.

2 Windows

For concreteness, we assume all downloads are stored in subdirectories of `c:/Downloads`, and installed to subdirectories of `c:/Gnu`, unless otherwise specified.

2.1 Environment Variables

Only the `HOME` environment variable needs to be declared in the Windows environment; others are defined in `~/.emacs` or `~/.bash_profile` (for GPS). `HOME` declares where `~/.emacs` and `~/.bash_profile` are found, and provides a place to store other configuration information.

To declare the `HOME` environment variable, navigate to `Control Panel | System | Advanced | Environment`. Set `HOME` to a directory of your choice, for example `e:\Stephe`.

2.2 Cygwin

1. From http://sources.redhat.com/cygwin/ click on "install cygwin now" to download `setup.exe` to directory `C:/Downloads/Cygwin`.

2. Create a desktop icon for `setup.exe`, so you can easily run it later to refresh Cygwin.

3. Run `setup.exe`. Follow the wizard; here are some of the settings you need:

   (a) install to `C:/Gnu/cygwin`

   (b) install for "all users"

   (c) default text file type "unix"

   (d) local package directory `C:/Downloads/Cygwin`

   (e) select `ftp://ftp.nas.nasa.gov`

   (f) select packages:

      i. base

      ii. Devel/cvs

      iii. Devel/make

      iv. Net/openssh

      v. Utils/cygutils

4. Edit `c:/Gnu/cygwin/etc/passwd` to set your home directory correctly. The last line should start with your login name, and end with `/home/<login>:/bin/bash` or something similar (seems to change each time I do this); change that to `/cygdrive/c/<your_home>:/bin/bash`. This is used by ssh.

5. People often have lots of stuff installed in Windows; consider specifying your PATH completely in `~/.bash_profile`:

```
PATH=/cygdrive/c/<your_home>/bin
PATH=$PATH:/cygdrive/c/Gnu/GNAT-5.01a/bin
PATH=$PATH:/cygdrive/c/texmf/miktex/bin
PATH=$PATH:/cygdrive/c/Gnu/cygwin/bin
PATH=$PATH:/cygdrive/c/Gnu/cygwin/usr/local/bin
PATH=$PATH:/cygdrive/c/winnt/system32

export PATH
```

2.2.1 Cygwin refresh

RedHat releases updates to Cygwin quite often. These usually fix bugs, add features, and only occasionally break things. Thus our policy is to keep up-to-date with the RedHat releases.

1. Before running setup, exit all cygwin processes.

2. Run `setup.exe` (via the desktop icon).
3. Leave everything defaulted from the previous install and hit next a lot.

3 LynxOS

See `lynx_4.0_notes.text` for basic Lynx OS setup and tools installation.

The rest of this document shows Windows commands; Lynx commands are similar, since it also provides a bash shell.

4 ssh

`ssh` is the secure shell, which is used for access to the CVS repository, and to machines over the network.

ssh-agent provides access to all machines on the network via one password. The setup process given here enables that access by distributing secure keys. It is correct for all operating systems.

1. Ask Stephe to get you an account on osgroup (the 582 CVS server), with access to:
   - /home/cvs/public/emacs
   - /home/cvs/public/ada
   - /home/cvs/gds
2. In an editor, add `export CVS_RSH=ssh` to `~/.bash_profile`.
3. In a bash shell:
   (a) `ssh-keygen -t rsa`
   This prompts for:
      - File in which to save the key. The default file is `~/.ssh/id_rsa`; keep it by hitting return.
      - Passphrase. This can be the same as your login password.

   (b) `scp ~/.ssh/id_rsa.pub osgroup.gsfc.nasa.gov:~/`

   (c) `ssh osgroup.gsfc.nasa.gov`

   (d) Change your password, via `passwd`.

   (e) if .ssh does not exist on osgroup:

      i. `mkdir .ssh`

ii. `chmod 755 .ssh`

(f) `cat id_rsa.pub >> .ssh/authorized_keys`

(g) `chmod 600 .ssh/authorized_keys`
owner read-write only

(h) `rm id_rsa.pub`

(i) `exit`

Note that the only way to change the passphrase is to run `ssh-keygen` again, which generates a new key, which must then be copied to osgroup again.

4.0.2 ssh-agent

All processes that need access to other machines must be started from a shell that is a child of the ssh-agent process. Thus, at system power up, the user must:

1. Launch a bash shell.
2. Type `ssh-agent bash`.
3. Type `ssh-add`, and provide the passphrase.
4. `ssh osgroup.gsfc.nasa.gov`
This verifies that a password is not required to access the osgroup account. If it is still required, then ssh setup was not successful.
5. Launch other tools (emacs, GPS, cvs).

5 Emacs

5.1 Windows

On Windows, we use the non-Cygwin version of Emacs, mainly because the GNAT compiler is also non-Cygwin.

1. Download `emacs-21.3-fullbin-i386.tar.gz` from ftp://ftp.gnu.org/pub/gnu/windows/emacs/latest to directory `C:/Downloads/Emacs`.
2. Install Emacs. In a bash shell:

```
cd c:/Gnu
mkdir Emacs
cd Emacs
tar zxf ../../Download/Emacs/emacs-21.3-fullbin-i386.tar.gz
```

5.2 Lynx

On Lynx, we use the provided binary, with patches (as described in `lynx_4.0_notes.text`).

5.3 Customizations

On all operating systems, we also use Stephe's customizations, which are stored in the branch CVS server.

Emacs is started from a bash shell that is under `ssh-agent`, so it can access the CVS repository transparently.

Paths shown are for Windows; for Lynx, replace `c:` with `~/`.

3. Install Stephe's customizations. In a bash shell under ssh-agent (`<user>` is replaced by your username on osgroup):

```
cd c:/Gnu/Emacs
mkdir site-lisp
cd site-lisp
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/public/emacs \
    checkout emacs_stephe
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/public/emacs \
    checkout emacs_stephe_site_lisp
cp emacs_stephe/example-windows.emacs ~/.emacs
c:/Gnu/Emacs/emacs-21.3/bin/runemacs.exe
```

   The last command starts Emacs.

4. Edit `~/.emacs` (select File — Open):
   (a) Replace all `<user>` with your login (select Edit — Search — Replace).
   (b) Edit other stuff as needed, to reflect your paths, etc.
5. Tell bash under Emacs to use a simpler prompt: create `~/.emacs_bash`, containing: `export PS1='\u@\h: '`.
6. Create a shell script `~/bin/emacs.sh` to start Emacs under ssh-agent (select File — Open to create a new file). For example:

```
# start emacs from ssh client with nice parameters
c:/Gnu/Emacs/emacs-21.3/bin/runemacs.exe --geometry=120x45-0+0
```

6 TeX

6.1 Windows

MikTeK is an implementation of TeX and LaTeX for for Windows. We use it for all documents.

1. Download `setup.exe` from http://www.miktex.org/setup.html to `C:/Downloads/TeX`.
2. Download MiKTeX packages. Run `setup.exe`. Title bar shows version 2.3.1223. Follow the prompts:
   (a) Select "Download only".
   (b) Select "small".
   (c) Select a repository: "USA" `ftp://tug.ctan.org/..` is good.
   (d) Set "local package repository" to `C:/Downloads/TeX/MiKTeX`.
   (e) Wait for download to complete.
3. Install standard MiKTeX packages. Run `setup.exe` again, follow the prompts:
   (a) Select "Install".
   (b) Select "small".
   (c) Select "Install a shared MiKTeX environment".
   (d) Set "local package repository" to `C:/Downloads/TeX/MiKTeX` (default is correct).
   (e) Set "Path to installation folder" to `C:/texmf` (note this is not under `Gnu`).
   (f) Set "Program folder name" to `MiKTeX`
   (g) Select "Create local TEXMF tree", path to root folder `c:localtexmf`.
   (h) Select "Don't incorporate existing TEXMF folder trees now".
   (i) Wait for installation to complete.
4. Install extra MiKTeX packages:
   (a) Run `Start | MikTex | MikTex Options`.
   (b) Select tab "Packages".
   (c) Under "Download Site", click "Change..."
      i. Select "Internet"
      ii. For the package repository, select "USA ftp://tug.ctan.org/..."

(d) In the package tree, select `Applications/Graphics/xypic`.

(e) Click "Apply", "ok"; the "MiKTeX Update" dialog opens. Wait for package to be downloaded and installed, Click "Close". Manuals for the xypic package are now in `c:/texmf/doc/generic/xypic/*.pdf`.

(f) Click "Show Catalogue"; the "MiKTeX Package Manager" opens.

(g) Select "fancyhdr", click "+" in toolbar, wait for installation. If it seems to hang, hit "cancel" and try again. Documentation for this package is in `c:/texmf/doc/latex/fancyhdr/fancyhdr.pdf`

5. Tell Windows to associate `.dvi` with Yap (Yet Another Previewer): select `Start | Programs | MiKTeX | DVI Viewer`.

6. Compile the LaTeX info manual

```
cd c:/texmf
makeinfo -o doc/latex/base/latex2e.info source/info/latex2e-help-texinfo/latex2e.texi
```

7. In `c:/texmf/doc/latex/base`, create file named `dir`, containing:

```
^_
File: dir Node: Top

* Menu:

* LaTeX2e: (latex2e).
```

Note that `^_` is one control character, produced in Emacs by `C-q _`.

8. In `~/.emacs` :

(a) `(add-to-list 'Info-default-directory-list "c:/texmf/doc/latex/base")`

(b) `c:/texmf/miktex/bin` to `exec-path`

6.2 Lynx

We don't use TeX under Lynx (at least for now). All documents are produced under Windows.

7 GNAT

7.1 Windows

7.1.1 Compiler

GNAT uses the Cygwin dll for the debugger, but installs a different version of the dll than installed by Cygwin. So we have to fix that.

1. Download from ACT FTP; access https://www.gnat.com/beta-customers/ using your web browser, enter our customer number and password; get customer number and password from Stephe. Individual files can be downloaded by clicking on desired file (the File Download dialog will appear). Do this, rather than selecting a group of files. If you select a group of files, it gives you a tar file with path information, which we don't want. Save to desired dowload area.

   Files to download, to directory `C:/Downloads/GNAT/5.01a`:

   (a) `gnat-5.01a-nt.exe`
   (b) `gps-1.3.0-nt.exe`
   (c) `gnat-5.01a-unw-docs.tar.gz`
   (d) `sources/asis-5.01a-src.tgz`
   (e) `tools/gnatwin-5.01a.exe`

2. Run `gnat-5.01a-nt.exe`. It will offer to uninstall any current GNAT; say "no", in case we need to back up a version.

   (a) Install to `c:\Gnu\GNAT-5.01a`

3. Run `gnatwin-5.01a.exe`

4. In bash:

```
cd c:/Gnu/GNAT-5.01a/bin
mv cyggnat.dll cyggnat-save.dll
cp ../../cygwin/bin/cygwin1.dll cyggnat.dll
```

5. In bash:

```
cd c:/Gnu/GNAT-5.01a/doc
tar zxf /cygdrive/c/Downloads/GNAT/5.01a/gnat-5.01a-unw-docs.tar.gz
mv gnat-5.01a-unw-docs/info/* info
rm -rf gnat-5.01a-unw-docs
```

6. In `c:/Gnu/GNAT-5.01a/doc/info`, create file named `dir`, containing:

```
^_
File: dir Node: Top

* Menu: The list of major topics begins on the next line.

* Gnu Compiler Collection: (gcc).
* Gnu Debugger:            (gdb).
* GNAT Reference:          (gnat_rm).
* GNAT User Guide:         (gnat_ugn).
* Gnu Visual Debugger:     (gvd).
```

   Note that `^_` is one control character, produced in Emacs by `C-q _`.

7. In `~/.emacs` :

   (a) `(add-to-list 'Info-default-directory-list "c:/Gnu/GNAT-5.01a/doc/info")`

   (b) Add `(concat gnu-dir "/GNAT-5.01a/bin")` to exec-path

7.1.2 ASIS

Ada Semantic Interface Specification (ASIS) is used in the `Auto_Text_IO` tool, which automatically generates text input and output packages for most Ada types. This is useful for debug test drivers.

1. In a bash shell:

```
cd c:/Gnu/Gnat-5.01a
tar zxf /cygdrive/c/Downloads/Gnat/5.01a/asis-5.01a-src.tgz
```

2. In `c:/Gnu/Gnat-5.01a/asis-5.01a-src/Makefile.stub`, change the value of `INSTALL_DIR` to `c:/Gnu/Gnat-5.01a/local`.

3. In a bash shell:

```
cd c:/Gnu/Gnat-5.01a/asis-5.01a-src/
make install
```

4. In `~/.emacs`, add an environment variable:

```
(setenv "GNAT_ASIS" "c:/Gnu/Gnat-5.01a/local/asis")
```

5. If not using Emacs, in `~/.bash_profile`, add an environment variable:

```
export GNAT_ASIS=c:/Gnu/Gnat-5.01a/local/asis
```

7.1.3 GPS

GPS provides an interface to the GNAT compiler and CVS. Thus it must be started from a process that is a child of `ssh-agent`, rather than from a desktop icon. So we set the environment variables it needs in `~/.bash_profile`, rather than in Windows.

GPS currently only runs on Windows, not Lynx.

1. run `gps-1.3.0-nt.exe`, install to `c:/Gnu/GPS-1.3.0`. This requires administrator rights, so you can't be logged in to a Windows domain when running this.

7.2 Lynx

GNAT for Lynx is a version behind GNAT for Windows. It is installed in a shared area, so only one user needs to install it.

7.2.1 Compiler

1. Download from ACT FTP; access https://www.gnat.com/beta-customers/ using your web browser, enter our customer number and password; get customer number and password from Stephe. Files to download, to directory `C:/Downloads/GNAT/3.16a1`:
   (a) `gnat-3.16a1-1-i386-elf-lynxos-bin.tar.gz`
   (b) `sources/asis-3.16a1-src.tgz`
2. We install in a non-standard location, to allow multiple versions of GNAT. We don't install as root, to simplify later installation of ASIS. In bash:

```
cd ~/Gnu
tar zxf ../Distrib/gnat-3.16a1-1-i386-elf-lynxos-bin.tar.gz
cd gnat-3.16a1-1-i386-elf-lynxos-bin
./doconfig
    choose option 3 (install in non-standard location)
    /Projects/GDS/GNAT-3.16a1-1
./doinstall
```

3. In `/Projects/GDS/GNAT-3.16a1-1/doc/gnat/info`, create file named `dir`, containing:

```
^_
File: dir Node: Top
```

```
* Menu: The list of major topics begins on the next line.

* GNAT Reference:          (gnat_rm).
* GNAT User Guide:         (gnat_ugn).
```

Note that `^_` is one control character, produced in Emacs by `C-q C-_`.

4. In `~/.emacs` :
   (a) `(add-to-list 'Info-default-directory-list "/Projects/GDS/GNAT-3.16a1-1/doc/gnat/inf`
   (b) Add `"/Projects/GDS/GNAT-3.16a1-1/bin"` to exec-path

7.2.2 ASIS

Note that we install ASIS in the shared area (`/Projects`, not `~/Projects`), so only one user needs to install it. ASIS depends on the compiler version.

1. In a bash shell:

   ```
   cd /Projects/GDS/GNAT-3.16a1-1
   tar zxf ~/Distrib/asis-3.16a1-src.tgz
   ```

2. In `/Projects/GDS/GNAT-3.16a1-1/asis-3.16a1-src/Makefile.stub`, change `INSTALL_DIR` to `/Projects/GDS/GNAT-3.16a1-1/local`.
3. In `/Projects/GDS/GNAT-3.16a1-1/asis-3.16a1-src/Makefile`, run `make install`.
4. In `~/.emacs`, add an environment variable:

   ```
   (setenv "GNAT_ASIS" (expand-file-name "/Projects/GDS/GNAT-3.16a1-1/local/asis"))
   ```

8 Misc tools

These tools are only occasionally used; you may not need them.

8.1 Elisp manual

The elisp manual is documentation on the Emacs Lisp language. It is not included in the standard Emacs distribution, but comes in handy sometimes.

1. Download `elisp-manual-21-2.8.tar.gz` from ftp://ftp.algx.net/pub/gnu/emacs/ (or another Gnu mirror) to `c:/Downloads/Emacs`.
2. In bash:

```
cd c:/Gnu
tar zxf ../Downloads/Emacs/elisp-manual-21-2.8.tar.gz
cd elisp-manual-21-2.8/
cp elisp elisp-* c:/Gnu/Emacs/emacs-21.3/info
cd c:/Gnu/Emacs/emacs-21.3/info
install-info elisp dir
```

Elisp now appears in the Emacs info menu.

8.2 GhostScript

GhostScript allows Emacs to print formatted text to a color printer.

1. Download ftp://mirror.cs.wisc.edu/pub/mirrors/ghost/AFPL/current/gs800w32.exe to `c:/Downloads/Misc`.
2. Run `gs800w32.exe`. This is a WinZip self-extractor; it brings up the GhostScript installer.
3. Install to `c:/Apps/gs`.
4. In `~/.emacs`, add:

```
;; Windows printing
(setq printer-name "Stephe's HP OfficeJet")
(setq ps-lpr-command "c:/Apps/gs/gs8.00/bin/gswin32c")
(setq ps-lpr-switches '("-q" "-dNOPAUSE" "-dBATCH" "-sDEVICE=mswinpr2"))
(setq ps-printer-name t) ; contrary to help for this item
```

Now the Emacs menu `File | Postscript Print Buffer` will print the current buffer to the printer, with headers, fonts and colorization. A Windows printer select dialog box will open first, letting you chose the printer.

8.3 ispell

This adds spell-checking to Emacs. The general web site for ispell is http://ficus-www.cs.ucla.edu/geoff/ispell.html. Ispell is available as source, so we have to compile it. Compiling it requires Cygwin and byacc.

1. Run Cygwin setup, add the following packages:
   (a) Devel/byacc
   (b) Devel/gcc-2

   Note that Devel/gcc may work, but I've only tested with Devel/gcc-2.
2. Download http://fmg-www.cs.ucla.edu/geoff/tars/ispell-3.2.06.tar.gz to `c:/Downloads/Misc`.
3. In bash:

```
cd c:/Gnu
tar zxf ../Downloads/Misc/ispell-3.2.06.tar.gz
cd ispell-3.2.06
cp local.h.samp local.h
chmod +w local.h
echo "#define USG" >> local.h
echo "#define CC g̈cc-2ï" >> local.h
echo "#define YACC b̈yaccï" >> local.h
make all
make install
mkdir -p /usr/local/bin
cp ispell.exe /usr/local/bin
mkdir -p /usr/local/lib
cp languages/american/americanmed.hash /usr/local/lib
```

4. In `~/.emacs`:
   (a) Add:

```
;; Ispell dictionary
(setenv "DICTIONARY" "/usr/local/lib/americanmed.hash")
```

   (b) Add `(concat gnu-dir "/cygwin/usr/local/bin")` to exec-path.

Now `M-$` will spell-check a word in emacs.

8.4 Maxima

Maxima is a symbolic algebra system; we use it to document derivations of algorithms. The primary web site is http://maxima.sourceforge.net/

1. Download `maxima-5.9.0.exe` from http://prdownloads.sourceforge.net/maxima/maxima-5.9.0.exe?download to `c:/Downloads/Misc`.
2. Run `maxima-5.9.0.exe`. Agree to the license, etc.

3. Install to `c:/Apps/Maxima-5.9.0`
4. Edit Maxima-5.9.0/bin/maxima:
   (a) Replace the first line `#/bin/sh` with `#!/bin/bash`
   (b) Replace line 32:

```
prefix=c:/msys/1.0/maxinstall
```

   with

```
prefix="`dirname \`dirname $0\``".
```

5. In `~/.emacs`, add `"c:/Apps/Maxima-5.9.0/bin"` to exec-path.

9 GDS source

The GDS source code is kept in the branch CVS server.

For Windows, in bash:

```
mkdir -p c:/Projects/GDS
cd c:/Projects/GDS
```

For Lynx, in bash:

```
mkdir -p ~/Projects/GDS
cd ~/Projects/GDS
```

For Lynx or Windows, in bash:

```
mkdir -p local/bin
mkdir auto_text_io
mkdir common
mkdir gpm
mkdir sdo
mkdir sal
mkdir makerules

cd auto_text_io
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/public/ada \
    checkout -d main Auto_Text_IO
cd ../common
```

```
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/gds \
    checkout -d main common
cd ../gpm
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/gds \
    checkout -d main gpm
cd ../sdo
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/gds \
    checkout -d main sdo
cd ../sal
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/public/ada \
    checkout -d main SAL
cd ../makerules
cvs -d :ext:<user>@osgroup.gsfc.nasa.gov:/home/cvs/public/ada \
    checkout -d main Makerules
```

where `<user>` is replaced by your username on osgroup.

For Windows, in Emacs, in `~/.emacs`, add environment variables:

```
(setenv "AUTO_TEXT_IO_ROOT" "c:/Projects/GDS/makerules/main")
(setenv "GDS_COMMON_ROOT" "c:/Projects/GDS/common/main")
(setenv "GNAT_ASIS" "c:/Projects/GDS/local/asis"))
(setenv "INSTALL_BIN" "c:/Projects/GDS/local/bin")
(setenv "MAKERULES_ROOT" "c:/Projects/GDS/makerules/main")
(setenv "SAL_ROOT" "c:/Projects/GDS/sal/main")

(setenv "GNAT_VERSION" "5.01a")
(setenv "OS_VERSION" "Windows_2000")
```

Also add `.../GDS/local/bin` to your path.

For Lynx, in Emacs, in `~/.emacs`, add environment variables:

```
(setenv "AUTO_TEXT_IO_ROOT" (expand-file-name "~/Projects/GDS/auto_text_io/main"))
(setenv "GDS_COMMON_ROOT" (expand-file-name "~/Projects/GDS/common/main"))
(setenv "GNAT_ASIS" (expand-file-name "~/Projects/GDS/local/asis"))
(setenv "INSTALL_BIN" (expand-file-name "/Projects/GDS/GNAT-3.16a1-1/local/bin"))
(setenv "MAKERULES_ROOT" (expand-file-name "~/Projects/GDS/makerules/main"))
(setenv "SAL_ROOT" (expand-file-name "~/Projects/GDS/sal/main"))

(setenv "GNAT_VERSION" "3.16a1")
(setenv "OS_VERSION" "Lynx_4.0")
```

Also add `(substitute-in-file-name "$INSTALL_BIN")` to your path.

For Windows GPS, in `~/.bash_profile`, add environment variables:

```
# environment variables for GPS
export SAL_ROOT=c:/Projects/GDS/SAL/main
export MAKERULES_ROOT=c:/Projects/GDS/Makerules/main
export GDS_COMMON_ROOT=c:/Projects/GDS/common/main

export ADA_PROJECT_PATH=$MAKERULES_ROOT:$SAL_ROOT/Build/Gnat_Debug;
export ADA_PROJECT_PATH=$ADA_PROJECT_PATH:\
$GDS_COMMON_ROOT/opentoken/build/x86_gnu_windows_test
```

`Auto_Text_IO` must be built first; it generates the `Text_IO` packages for SAL, which are used in GDS debug executables. Then GDS can be built.

In file `c:/Projects/GDS/auto_text_io/main/Build/x86_gnu_windows_release/Makefile`, run `make -r all install`

In file `c:/Projects/GDS/common/main/Build/x86_gnu_windows_test/Makefile`, run `make -r all`.

GDS code is now built.